

EE 354
C51 Program Format

September 16, 2016

C51 language programs for the 8051 should be turned in using the following format:

```
// Your Name
// Assignment number
// Date
//
// Header describing what program does
//
#include <at89c51cc03.h>      // CC03 library file
//
Declare any global variable here
//
Put in subprogram prototypes here
//
//
void main(void)              //This is the start of the main program
{Declare all variables here
  ...
  main code goes here
  main code must be a perpetual loop or
  end in a while(1);
}
//
Put in subprograms here
A complete example is given on the following page.
```

```

//Your name
//Assignment umpty-ump
//September 16, 2016
//
//RTClock.c
// This program implements a real time clock in c for the
// 8031. The interrupt is set for 50,000 ticks. At 12MHz
// this is 50 msec. At 6MHz this is 100msec = 0.1 second. The clock
// increments a minutes variable each 10 interrupts.

#include <at89c51cc03.h>
//Clock variable are global so that they can be accessed anywhere
// in the program
unsigned char seconds;
unsigned char minutes;
unsigned char TenthsCounter;
//Function prototypes
void OneMinute(void);
//
void main(void)
    {TMOD = 0x01;    //Timer 0 mode = not gated, internal clock, 16 bit,
                  // no auto reload
    TH0 = 0x3c;    //Timer 0 high and low byte. Interrupt
    TL0 = 0xb0;    // occurs when timer overflows on up count
    TR0 = 1;      //Timer 0 run control bit in TCON
    ET0 = 1;      //Timer 0 interrupt enable
    EA = 1;      //Global interrupt enable
    while(1);    //Wait here for interrupt
    }
//
//Interrupt service routine uses register bank 1

void OneMinute(void) interrupt 1 using 1
    {TH0 = 0x3c;    //Reload the count register
    TL0 = 0xb0;
    TenthsCounter++;    //Interrupt at msec/10
    if(TenthsCounter > 9)    //For each 10 update seconds
        {TenthsCounter = 0;
        seconds++;
        if(seconds > 59)    //For each 60 seconds update minutes
            {seconds = 0;
            minutes++;    //minutes can overflow
            }
        }
    }
}

```